

## CLAIMS

What is claimed is:

1. A method comprising:
  - determining, by a first program, an attribute of a first functional unit by referencing a virtual memory address, the first functional unit comprising a first processor and a random access memory (RAM) coupled to the first processor in a computer system, and the first program executing in the first functional unit;
  - determining, by a second program, an attribute of a second functional unit by referencing the virtual memory address, the second functional unit comprising a second processor and a RAM coupled to the second processor in the computer system, and the second program executing in the second functional unit; and
  - wherein the referencing the virtual memory address by the first program provides a pointer to an attribute stored in the RAM of the first functional unit, and wherein the referencing the virtual memory address by the second program provides a pointer to an attribute stored in the RAM of the second functional unit.
2. The method as defined in claim 1 wherein, prior to the determining steps, the method further comprises:
  - saving a replicated portion of an operating system program and the attribute of the first functional unit to a read-only portion of the RAM of the first functional unit; and
  - saving a replicated portion of an operating system program and the attribute of the second functional unit to a read-only portion of the RAM of the second functional unit.
3. The method as defined in claim 1 wherein determining an attribute of a first functional unit further comprises determining a functional unit identification number.

4. The method as defined in claim 1 wherein determining an attribute of a first functional unit further comprises determining low and high physical address of the RAM of the first functional unit.
5. The method as defined in claim 1 wherein determining an attribute of a first functional unit further comprises determining a list of input/output devices local to the first functional unit.
6. A computer system comprising:
  - a first processor coupled to a first random access memory (RAM), the first processor and first RAM forming a first resource affinity domain (RAD);
  - a second processor coupled to a second RAM, the second processor and second RAM forming a second RAD, and wherein the second processor is coupled to the first processor;
  - a RAD specific attribute of the first RAD along with a replicated portion of an operating system stored in the first RAM; and
  - a RAD specific attribute of the second RAD along with a replicated portion of the operating system stored in the second RAM.
7. The computer system as defined in claim 6 further comprising:
  - wherein the replicated portion of the operating system, when executing in the first RAD, reads the RAD specific attribute of the first RAD by reference to a virtual memory address; and
  - wherein the replicated portion of the operating system, when executing in the second RAD, reads the RAD specific attribute of the second RAD by reference to the virtual memory address.
8. The computer system as defined in claim 6 further comprising:
  - wherein the RAD specific attribute of the first RAD is a RAD identifier; and

wherein the operating system program stored in the first RAM, when executed by the first processor, determines the RAD within which the operating system program is executed by reading the RAD identifier from the first RAM.

9. The computer system as defined in claim 6 wherein the RAD specific attribute is a RAD identifier.

10. The computer system as defined in claim 9 wherein each replicated operating system program, when executed by the processors in its RAD, uses the RAD identifier to determine a local RAM for memory allocation.

11. The computer system as defined in claim 9 wherein each replicated operating system program, when executed by the processor in its RAD, uses the RAD identifier for scheduling a program stored in local RAM.

12. A computer readable media comprising an executable program that, when executed, implements a method comprising:

reading a functional unit identifier from a random access memory (RAM) coupled to a processor in which the program executes; and thereby determining within which functional unit, identified by the functional unit identifier, the program is executing.

13. The computer readable media as defined in claim 12 wherein the executable program further comprises allocating memory from RAM within the functional unit, identified by the functional unit identifier, to a program executing on the processor in the functional unit.

14. The computer readable media as defined in claim 12 wherein the executable program further comprises scheduling a program to execute on the processor in the functional unit.

15. A computer system comprising:  
a first means for executing programs coupled to a first means for storing programs and instructions, the first means for executing and means for storing forming a first functional unit;  
a second means for executing programs coupled to a second means for storing programs and instructions, the second means for executing and means for storing forming a second functional unit;  
an attribute of the first functional unit along with a replicated portion of an operating system stored in the first means for storing; and  
an attribute of the second functional unit along with a replicated portion of the operating system stored in the second means for storing.
16. The computer system as defined in claim 15 further comprising:  
wherein the replicated portion of the operating system, when executing in the first functional unit, reads the attribute of the first functional unit by reference to a virtual memory address; and  
wherein the replicated portion of the operating system, when executing in the second functional unit, reads the attribute of the second functional unit by reference to the virtual memory address.
17. The computer system as defined in claim 15 further comprising:  
wherein the attribute of the first functional unit is a functional unit identifier;  
and  
wherein the operating system program stored in the first functional unit, when executed by the first means for executing, determines the functional unit within which it is executed by reading the functional unit identifier from the first means for storing.
18. The computer system as defined in claim 15 wherein the attribute is a functional unit identifier.

19. The computer system as defined in claim 18 wherein each replicated operating system program, when executed by the means for executing in its functional unit, uses the functional unit identifier to determine a local means storing for allocation.

20. The computer system as defined in claim 18 wherein each replicated operating system program, when executed by the means for executing in its functional unit, uses the functional unit identifier for scheduling a program stored in a local means for storing.